

**ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД УКООПСІЛКИ
«ПОЛТАВСЬКИЙ УНІВЕРСИТЕТ ЕКОНОМІКИ І ТОРГІВЛІ»**

**НАВЧАЛЬНО-НАУКОВИЙ ІНСТИТУТ БІЗНЕСУ ТА СУЧАСНИХ ТЕХНОЛОГІЙ
ФОРМА НАВЧАННЯ ДЕННА
КАФЕДРА МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ ТА СОЦІАЛЬНОЇ
ІНФОРМАТИКИ**

Допускається до захисту

Завідувач кафедри _____ О.О. Ємець

«_____» _____ 2020 р.

**ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО МАГІСТЕРСЬКОЇ РОБОТИ**

на тему:

**РОЗРОБКА ТРЕНАЖЕРУ З ТЕМИ «ВИЯВЛЕННЯ АНОМАЛЬНИХ
СПОСТЕРЕЖЕНЬ ЗА ДОПОМОГОЮ КРИТЕРІЮ ТОМПСОНА» ДИСЦИПЛІНИ
«АНАЛІЗ ДАНИХ І ПРИКЛАДНІ ПАКЕТИ СТАТИСТИЧНОЇ ОБРОБКИ»**

зі спеціальності 122 «Комп'ютерні науки»

Виконавець роботи: Кривошей Олег Сергійович

_____ «___» _____ 2020 р.

Науковий керівник: к.ф.-м.н., доц. Олексійчук Юрій Федорович

_____ «___» _____ 2020 р.

ПОЛТАВА 2020 р.

ЗМІСТ

ВСТУП	3
1. ПОСТАНОВКА ЗАДАЧІ	6
1.1 Постановка задачі розробки тренажера	6
2. ІНФОРМАЦІЙНИЙ ОГЛЯД	8
2.1 Необхідність та актуальність роботи	8
2.2 Огляд робіт, де розглянуте аналогічне до теми роботи завдання	9
2.3 Позитивні аспекти оглянутих робіт	11
2.4 Вади розробок з оглянутих робіт	11
3. ТЕОРЕТИЧНА ЧАСТИНА	13
3.1 Розв’язування задачі з виявлення аномальних спостережень за допомогою методу Томпсона в табличному редакторі MS-Excel	13
3.2 Алгоритм роботи тренажера	17
3.3 Розробка блок-схеми, яка підлягає програмуванню	25
3.4 Обґрунтування вибору програмних засобів для реалізації завдання роботи	29
4. ПРАКТИЧНА ЧАСТИНА	31
4.1 Опис процесу програмної реалізації	31
4.2 Опис програми	35
ВИСНОВКИ	44
СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ	46
ДОДАТОКИ	50
Додаток А. Компакт-диск з тренажером	50
Додаток Б. Код програми	51

**ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ,
ТЕРМІНІВ**

Умовні позначення, символи, скорочення, терміни	Пояснення умовних позначень, скорочень, символів
MS Excel	Табличний процесор, програма для роботи з таблицями, створена корпорацією Microsoft
E – learning	Електронне навчання
JavaScript	Об'єктно-орієнтована мова програмування
Electron	Фреймворк, що дозволяє розробляти рідні графічні застосунки для настільних операційних систем за допомогою веб-технологій.
Фреймворк	Інфраструктура програмних рішень, що полегшує розробку складних систем.
Node.js	Платформа з відкритим кодом для виконання високопродуктивних мережових застосунків, написаних мовою JavaScript.
UI	Засіб зручної взаємодії користувача з інформаційною системою

ВСТУП

Зараз, як ніколи раніше, гостро стоїть питання дистанційного навчання. Зміни, які зазнало наше суспільство за останні роки у галузі освіти та науки є викликом для навчальних закладів. Головною метою цього виклику є постійний пошук нових шляхів в напрямку підготовки людей до життя в сучасному суспільстві. Світовий розвиток науки і техніки, який зараз відбувається - це є ніщо інше, як інформаційний вибух сучасних технологій у всіх сферах людської діяльності, який, передусім, ставить завдання для освіти вплинути на студента так, щоб він впорався з постійно зростаючими вимогами, які ставляться до нього.

Дистанційне електронне навчання – це метод, який встановлює прямий контакт між студентом та викладачем з використанням Інтернет технологій. Навчання на відстані означає оволодіння наукою самостійно, без фізичної участі викладача, як у традиційній формі розуміння цього поняття. Кожен учасник цього процесу визначає мету навчання самостійно. Змістом уроків дистанційного навчання є створення віртуальних середовищ, так званих “он-лайн груп”. Учасники цих груп повинні використовувати пошук інформації з усіх можливих Інтернет джерел, які поділяються за спільною тематикою.

Замість вивчення прикладів, описаних у довідниках, студенти які навчаються дистанційно, мають можливість працювати з насимульованими програмами та он-лайн тренажерами, а також працювати в командах та ділитися своїми ідеями через Інтернет.

Основні переваги дистанційної освіти:

1. Навчання може бути проведено не залежно від часу та місця.
2. Наявність забезпечення можливості двостороннього спілкування (вчитель - студент, студент - студент).
3. Можливість використання різних освітніх технологій та засобів масової інформації для зв'язку.

З вищевикладеного матеріалу випливає, що дистанційна освіта є мультимедійною формою самонавчання, яка забезпечує нові освітні можливості та підтримує освітні послуги для всіх її учасників. Основна відповідальність за хід і результати навчання лежить на студентах, які відокремлені від викладачів.

Дистанційне електронне навчання можна розподілити за формами:

1. Безпосереднє вивчення (незалежне, асинхронне)

Воно полягає в своєчасному засвоєнні дидактичного матеріалу студентами без контакту з викладачем в реальному часі на основі симуляцій та онлайн тренажерів.

2. Викладання під керівництвом викладачів (синхронне навчання)

Ця форма навчання передбачає звичайні дії викладача в процесі управління викладанням та реальну взаємодію між викладачем та студентом. Під час навчання, студенти можуть спілкуватися між собою. Це відбувається у віртуальних аудиторіях.

Ці два типи викладання в поєднанні надають максимально високий рівень сприйняття та засвоєння інформації студентом, але пріоритетнішим для навчальних закладів є асинхронний вид навчання. Так як цей вид навчання не залежить від викладачів та не потребує їх постійної участі. Реалізацією цієї форми дистанційного навчання є написання різних програмних комплексів. Для цієї задачі найкраще підходять, так звані, програми-тренажери та симулятори .

Самостійне вивчення теми «Виявлення аномальних спостережень за допомогою критерію Томпсона» дисципліни «Аналіз даних і прикладні пакети статистичної обробки» є досить складною справою. Наявність програми-тренажеру могло б суттєво допомогти студентам в освоєнні даної теми. Такий тренажер ще не був розроблений, хоча тема не є простою в розумінні. Отже, розробка тренажеру з теми «Виявлення аномальних спостережень за допомогою критерію Томпсона» дисципліни «Аналіз даних і прикладні пакети статистичної обробки» є актуальною. Перевагою перед іншими тренажерами повинна бути: можливість крос-платформенного запуску тренажеру та прикладу MS- Excel на будь-якій операційній системі, а також можливість

одночасного використання великого обсягу навчальної інформації будь - якою кількістю студентів.

Актуальність – перспектива впровадження сучасних методів навчання в тему: «Виявлення аномальних спостережень за допомогою критерію Томпсона» дисципліни «Аналіз даних і прикладні пакети статистичної обробки» завдяки використанню Інтернет технологій та програмних компонентів комп'ютерів

Мета роботи – розробити тренажер, котрий буде навчати студентів розв'язувати задачі по виявленню аномальних спостережень за допомогою прикладного пакету програм MS-Excel.

Об'єкт роботи – розробка тренажеру для систем дистанційного навчання.

Предмет роботи предметом роботи є програма-тренажер для навчання розв'язуванню задач по виявленню аномальних елементів за допомогою критерію Томпсона в програмному пакеті MS-Excel для дистанційного навчального курсу «Аналіз даних і прикладні пакети статистичної обробки»

Методи роботи – дипломну роботу було розроблено з використанням методу аналізу даних, а саме методу Томпсона. Для створення програмами було використано мову програмування JavaScript. Також було використано JavaScript бібліотеку «Sweet Alert 2» та JavaScript фреймворки «Electron» та «jQuery».

1. ПОСТАНОВКА ЗАДАЧІ

1.1 Постановка задачі розробки тренажера

В цій магістерській роботі головною задачею є розробка тренажеру з теми: «Виявлення аномальних спостережень за допомогою критерію Томпсона» для дистанційного курсу «Аналіз даних і прикладні пакети статистичної обробки», а також його впровадження в систему «Moodle»

Створення комп'ютерних програм виконується в декілька основних етапів:

1. Аналіз проблеми – розпочинається з відповіді на питання, що потрібно вирішити. Ретельно формується запис проблеми та вимоги до нової програми;
2. Пропонування рішення – розгляд всіх можливих варіантів, як можливо вирішити проблему. Шукаючи рішення, потрібно виходити з того, що відомо про дану область, які засоби можливо використати у своєму розпорядженні для вирішення поставленого завдання. Результатом цієї роботи буде алгоритм, тобто послідовність кроків;
3. Реалізація – переклад запланованого алгоритму на конкретну мову програмування, яку розуміє і може виконувати комп'ютер;
4. Технічне обслуговування - під технічним обслуговуванням мається на увазі виявлення та виправлення прихованих помилок, адаптація програмного забезпечення до мінливих вимог користувача.

З цього випливає, що основними завданнями під час розробки є:

1. Вибір основного напрямку розробки ПЗ під операційну систему;
2. Створення Алгоритму роботи програми-тренажеру;
3. Складання блок-схеми до алгоритму з врахуванням всіх можливих варіантів роботи;
4. Вибір мови програмування відповідно до зручного користування на основних операційних системах;
5. Програмна реалізація тренажеру;

6. Перевірка та тестування програми на відповідність та правильність роботи.

Програмне забезпечення має відповідати таким характеристикам як: модульність, масштабованість та незалежність від платформи. Також програмний продукт може бути реалізований на базі таких Інтернет-браузерів, як: Mozilla Firefox та Google Chrome (chromium подібних браузерів).

Основні вимоги до програмного продукту:

1. Під час роботи програми, студент повинен мати можливість побачити та зрозуміти базовий принцип роботи методу Томпсона..
2. Після вибору відповіді або при введенні даних, необхідно реалізувати механізм перевірки даних на відповідність та, у випадку помилки, створити метод виведення повідомлення про не вірну відповідь.
3. Створити механізм можливості повернення до попереднього кроку роботи тренажеру, або ж пропуску кроку.

2. ІНФОРМАЦІЙНИЙ ОГЛЯД

Останнім часом з'явилося дуже багато публікацій присвячених навчальним тренажерам і дистанційному навчанню [5-17]. Програми-тренажери, які були розроблені, використовуються для тренування навичок пов'язаних з розв'язанням типових задач. За допомогою таких тренажерів можливо працювати з індивідуальним підходом до навчання з кожним студентом. Таким чином, перевагами даної форми навчання є:

- можливість завчасного створення завдання;
- можливість покрокового вирішення завдання;
- можливість навчання студента з конкретної теми в буд-який зручний час;
- вчасне надання вірної відповіді чи можливості знайти вірну відповідь;
- можливість мати незліченну кількість повторень конкретної теми;
- відсутність необхідності контролю зі сторони викладача.

2.1 Необхідність та актуальність роботи

Сучасне суспільство розвивається вражаючими темпами. Аналіз тенденцій, напрямів і характеру розвитку економіки сучасного світу показує та підтверджує, що сьогодні найбільш перспективною парадигмою розвитку людства виступає модель „сума високих технологій”, яка базується на сучасній, що розвивається надзвичайно швидкими темпами, інформаційно-комунікаційній інфраструктурі. Реалізуючи цю модель, навчальні заклади створюють умови для формування нового технологічного укладу, який передбачає інтенсивний взаємозв'язок та взаємозбагачення різних технологічних напрямів.

Розвиток сучасної освіти в Україні базується на використанні комп'ютерної техніки, інформаційних технологій та комп'ютерних комунікацій у навчальній діяльності. Створення новітнього електронного середовища для навчання є ніщо інше, як розвиток навчальної системи. Перед освітою постає задача – сформувати конкурентоспроможну особистість, яка буде успішною в електронному середовищі.

Студенти, які працюють в сучасному електронному середовищі, не прив'язані до робочого місця та мають можливість пересуватися разом з ним. Створення такого середовища кардинально змінює парадигму надання знань. Ця модель орієнтована на принципи «вчитися завжди і всюди». Упровадження електронного навчання (E – learning) в українську систему освіти є актуальним та дозволяє здійснювати освіту на відстані, а також експортувати вітчизняні освітні послуги за кордон.

Основна мета дистанційного навчання є надання можливість отримання якісних знань всім бажаючим, в будь-який час.

Дистанційна освіта є однією з найперспективніших форм освіти зараз. Дистанційна освіта це форма заочної освіти, але, в той самий час, дистанційна освіта значною мірою є більш ефективна, ніж стандартна заочна освіта. Тому дистанційна освіта передбачає набагато більше спілкування між студентом і викладачем.

Сучасний етап розвитку інформаційної освіти спрямований на подальше підвищення якості освіти, забезпечення конкурентоспроможності на світовому ринку освітніх послуг, а також її інтеграцію у світовий освітній простір. Він передбачає реалізацію принципів відкритої освіти, підпорядкований сучасним освітнім парадигмам людиноцентризму та рівного доступу до якісної освіти. Його характеризують суттєві цільові та змістово-технологічні зміни, що системно відбуваються в освітніх системах, як на рівні окремих комп'ютерно орієнтованих засобів навчання. Навчальні ресурси істотно розгортають предметний простір застосувань комп'ютерно орієнтованих засобів освітнього призначення.

2.2 Огляд робіт, де розглянуте аналогічне до теми роботи завдання

Під час написання магістерської роботи були розглянуті тренажери студентів попередніх років, а саме Томченка О.В, Русіна В.С, Сивокія О.Ю, а також Мельницького Я.В та Спорніка С.С.

Тренажер Томченка О.В. був розроблений на мові програмування Java, з теми: “Перший алгоритм Гоморі” в дистанційному навчальному курсі “Методи оптимізації

та дослідження операцій”. На початку роботи програми-тренажера, користувачу виводиться інформація про розробника, місце розробки та тему роботи. Під час роботи тренажера, програма контролює вірність кожної відповіді користувача. У випадку невірної відповіді або не коректного вводу даних, виводиться вікно про помилку, а також підказка.

Тренажер Русіна В.С. – Програмна реалізація елементів тренажеру з теми "Аналіз алгоритму сортування вставками" дисципліни "Аналіз алгоритмів" [10]. Загальний принцип роботи цього тренажера суттєво відрізняється від попереднього тренажеру. Він написаний на мові програмування JavaScript, що є важливою перевагою, так як ця мова належить до інтерпретованих. Також в даному розробленому тренажері використовується графічний інтерфейс, написаний на мові розмітки сторінки HTML та каскадна таблиці стилів CSS. Вельми зручним елементом цього тренажеру є його можливість використання історії виконаних дій під час роботи з ним. Окрім цього, зручність тренажеру вказано також в тому, що цей тренажер адаптований до роботи з ним за різними пристроями (телефонами, планшетами, комп'ютерами).

Тренажер Сивоконя О.Ю. [16] був виконаний у вигляді Java-аплету на мові програмування Java, з теми "Логіка висловлювань" дистанційного навчального курсу "Математична логіка" та розробка його програмного забезпечення. На початку роботи тренажера, відображаються три кнопки з вибором конкретного завдання, чи то тести, чи то практичне розв'язання задач, чи то розглядання задачі з поясненнями. Також однією з позитивних рис цього тренажеру є можливість роботи на мобільних девайсах під керуванням операційної системи Android.

Тренажер Мельницького Я.В. з теми: «Розробка тренажеру для вивчення основ мови програмування Python» [17] написаний на платформі JavaFX. Цей тренажер використовує мову розмітки користувацького інтерфейсу FXML. Після натиснення кнопки розпочати на першому вікні тренажеру, студент може вибрати з якого саме питання він хоче розширити свої знання в пізнанні мови програмування

Python, і це є однією з позитивних рис цього тренажеру. Також позитивною рисою цього тренажеру є його не одноманітність завдань.

Тренажер Спорніка С.С. з теми: «Частинна кореляція» дисципліни «Аналіз даних і прикладні пакети статистичної обробки» [13] також розроблений на мові програмування Java. Даний тренажер працює по схожому принципу, що й тренажер Томченка О.В.

2.3 Позитивні аспекти оглянутих робіт

1. У всіх проектах були використані крос-платформенні мови програмування, що в свою чергу дає змогу впроваджувати їх в дистанційний курс, за потреби;
2. В усіх переглянутих роботах простий та інтуїтивно зрозумілий інтерфейс.
3. Всі наведені тренажери, в тій чи іншій мірі, дають можливість отримати навички та знання з теми, на які вони покладені;
4. Деякі тренажери виконані у вигляді десктопних програм, завдяки цьому вони дозволяють пройти тест без доступу до мережі інтернет;
5. Наведені тренажери полегшують роботу викладача та дають можливість студенту більше розібратися в предметі, який він вивчає.
6. В тренажерах Мельницького Я.В та Сивоконя О.Ю. є можливість вибору кроку при початку роботи програми.
7. В деяких тренажерах використано не тільки мову програмування, а й мову розмітки користувацького інтерфейсу.
8. Тренажер Сивоконя О.Ю. максимально адаптований під одну з найбільш популярних операційних систем для телефону та планшету – Android.
9. Деякі тренажери розроблені не тільки на чистій мові програмування, але й з використанням фреймворків до цієї мови програмування.

2.4 Води розробок з оглянутих робіт

1. У всіх тренажерах одноманітні методи рішення завдання та самі завдання;

2. Не всі з наведених тренажерів мають можливість пропуску кроку і повернення назад;
3. В деяких тренажерах відсутній сучасний UI дизайн;
4. Деякі тренажери не мають десктопного, незалежного застосунку, а це значить, що вони можуть використовуватися лише, як влаштований елемент для відображення та роботи на сайті або у веб-браузері .
5. В тренажері Томченка О.В підказка є малоінформативною.

3. ТЕОРЕТИЧНА ЧАСТИНА

3.1 Розв'язування задачі з виявлення аномальних спостережень за допомогою методу Томпсона в табличному редакторі MS-Excel

В програмному пакеті MS OFFICE запусимо програму MS Excel та заносимо вихідні данні задачі на вільний аркуш (рис. 3.1).

	A	B
1	n	x
2	1	1
3	2	-1,4
4	3	3
5	4	2,3
6	5	1,5
7	6	1,7
8	7	1,1
9	8	1,5
10	9	4,9
11	10	2,3
12	11	1,6
13	12	0,9
14	13	0,7
15	14	0,9
16	15	0,5
17	16	1,4
18	17	2,9
19	18	2,2
20	19	2

Рисунок 3.1 – Данні задачі

Знаходимо середнє вибіркве зі списку елементів «x» (рис. 3.2).

fx =AVERAGE(B2:B20)		
G	H	I
Середнє вибіркве =		1,631578947

Рисунок 3.2 – Середнє вибіркве.

Далі знаходимо середнє стандартне відхилення зі списку елементів «x» (рис. 3.3).

Як відомо, всі формули в MS Excel розпочинаються із знаку “ = ” (рис. 3.5). Вибираємо комірку для розрахунку елементу $Z(\max)$ та записуємо до неї формулу, яка обчислює значення $Z(\max)$. Це значення можна розрахувати за допомогою вбудованої функції MS-Excel, MAX(). Нехай це буде комірка H4.

The screenshot shows the MS Excel interface. The formula bar at the top displays $=\text{MAX}(C2:C20)$. Below it, the spreadsheet shows a column of values in cells C2 through C20. The value 3.268421053 is highlighted in cell H4, indicating it is the maximum value found in the range C2:C20.

	C
1	z
2	0,631578947
3	3,031578947
4	1,368421053
5	0,668421053
6	0,131578947
7	0,068421053
8	0,531578947
9	0,131578947
10	3,268421053
11	0,668421053
12	0,031578947
13	0,731578947
14	0,931578947
15	0,731578947
16	1,131578947
17	0,231578947
18	1,268421053
19	0,568421053
20	0,368421053

Рисунок 3.5 – Знаходження елементу « $Z(\max)$ ».

Далі ми знаходимо $T(n)$, $t_{(n-2)}$ за формулами (3.1), (3.2) відповідно, де $x_{i(n)}$ – елемент в якого елемент z дорівнює $Z(\max)$, $\bar{x}(n)$ – середнє вибіркве, $S(n)$ – вибіркве стандартне відхилення, а n – кількість елементів « x ».

$$T(n) = \frac{x_{t(n)} - \bar{x}(n)}{S(n)} \quad (3.1)$$

$$t_{(n-2)} = \frac{\sqrt{n-2}T(n)}{\sqrt{n-1-T^2(n)}} \quad (3.2)$$

Таким чином, ми можемо знайти елемент t_{kr} за допомогою вбудованої функції MS-Excel, TINV(), в яку першим елементом буде вказано кількість елементів «х», а другим елементом буде рівень значимості, який відомо з умови і в даному випадку він дорівнює 0,1.

Знаючи такі елементи, як t_{kr} та $t_{(n-2)}$ можна визначити чи завершується дія алгоритму Томпсона. Завершення роботи алгоритму спрацьовує тільки тоді, коли виконується умова $t_{kr} > t_{(n-2)}$. Як це виглядає в програмі MS-Excel можна побачити на рисунку (рис. 3.6).

	F	G
9	n	19
10	T(n)=	2,588363931
11	t(n-2)=	3,17470537
12	t _{kr} =	1,739606726
13	кінець(0) чи далі(1)=	1
14	Продовжуємо	

Рисунок 3.6 – Прирівняння елементів t_{kr} та $t_{(n-2)}$.

В цьому випадку, коли умова не виконалась, потрібно повторювати роботу методу з початку враховуючи те, що тепер потрібно вилучити зі списку той елемент у якого z дорівнювало $Z(\max)$.

Як виглядає останній крок порівняння для конкретного даного прикладу можна побачити на рисунку (рис. 3.7).

	F	G
108	n	15
109	T(n)=	-1,606024937
110	t(n-2)=	-1,713475522
111	t _{kr} =	1,770933396
112	кінець(0) чи далі(1)=	0
113	Дія методу завершена	

Рисунок 3.7 – Прирівняння елементів t_{kr} та $t_{(n-2)}$ в останньому кроці роботи алгоритму.

3.2 Алгоритм роботи тренажера

На початку роботи тренажеру перед студентом відкривається вікно з двома кнопками та основною інформацією про тренажер. Одна з двох кнопок використовується для отримання базової інформації про метод Томпсона. Інша кнопка, котра містить на собі надпис «Розпочати», використовується для початку тренування.

В програмі-тренажері використовуються кроки з питаннями в яких є тільки одна вірна відповідь, яку має визначити сам студент. Також є кроки з введенням даних з клавіатури.

У випадках, якщо буде вибрано вірну відповідь, тренажер виведе інформацію на екран про вірність вибору і потім перейде до наступного питання, інакше буде виведена інформація на екран про невірний вибір та надана підказка.

Крок 1.

На початку роботи тренажера, перед студентом відкривається вікно, на якому виводиться питання й перелік можливих відповідей.

Далі питання: «Аномальним спостереженням називається?»

- a) - називається знаходження та ідентифікація елементів, подій або спостережень, що не відповідають очікуваній поведінці або іншим елементам набору даних;
- b) - називається знаходження та ідентифікація елементів, подій або спостережень, що відповідають очікуваній поведінці (патернам) або іншим елементам набору даних;
- c) - називається знаходження відповідностей між очікуваними подіями та набором даних суми всіх елементів.

Вірна відповідь – a)

Крок 2.

Перед студентом відкривається наступне вікно, де з'являється наступне завдання.

Заповніть таблицю. Впишіть значення N та X у відповідну клітинку. Значення X - це вибірка, а N - це номер елемента вибірки.

1	1
2	-1,4
3	3
4	2,3
5	1,5
6	1,7
7	1,1
8	1,5

Таблиця 3.1 – Завдання другого кроку.

Вірна відповідь буде такою: (таб. 3.2)

N	X
1	1
2	-1,4
3	3
4	2,3
5	1,5
6	1,7
7	1,1
8	1,5

Таблиця 3.2 – Відповідь на завдання другого кроку.

Крок 3.

Перед студентом відкривається наступний крок завдання, де необхідно ввести вірну відповідь з клавіатури, з точністю два знаки після коми.

«Визначте середнє вибіркєве з даної X вибірки. »

$X = \{1,3,1,5,1,1,4,9,1,6,0,7,0,5,2,9,-1,4,2,3,1,7,1,5,2,3,0,9,0,9,1,4,2,2\}$

Вірна відповідь – 1,611

Крок 4.

Далі питання: «В табличних редакторах типу MS Excel за допомогою якої функції рахується вибіркєве стандартне відхилення? $S(n)$ »

а) -STDEV() / СТАНДОТКЛОН();;

б) -STMEAN() / СТАНДВИБ();

с) - SLUBN()/ СЛУБОТКЛОН().

Вірна відповідь – а)

Крок 5.

Перед студентом відкривається наступне вікно, де з'являється наступне завдання.

Заповніть таблицю , відповідно до випадку з такими даними: Рівень значимості = **0,1**; Середнє вибіркове відхилення = **1,26273**; Середнє вибіркове = **1,63157**.

S(n)	
x(n)	
P	

Таблиця 3.3 – Завдання п'ятого кроку.

Вірна відповідь буде такою: (таб. 3.4)

S(n)	1,26273
x(n)	1,63157
P	0,1

Таблиця 3.4 – Відповідь на завдання п'ятого кроку.

Крок 6.

Далі студенту виводиться на екран наступне завдання.

Вкажіть пропущене число в стовпчик “**Z**”, враховуючи що кожен елемент “**Z_i**” знаходиться за формулою **Z_i = ABC(x_i - x(n))**

Середнє вибіркове		1,631
n	x	z
1	1	0,631
2	-1,4	3,031
3	3	
4	2,3	0,668
5	1,5	0,131

Таблиця 3.5 – Завдання шостого кроку.

Вірна відповідь буде такою: (таб. 3.6)

Середнє вибіркове		1,631
n	x	z
1	1	0,631
2	-1,4	3,031
3	3	1,369
4	2,3	0,668
5	1,5	0,131

Таблиця 3.6 – Відповідь на завдання шостого кроку.

Крок 7.

Виберіть найбільше число в стовпчику “Z” та впишіть його в пусту клітинку “**Z(max)**”, в пусту клітинку “**Xi(n)**” впишіть елемент “Xi”, який знаходиться навпроти найбільшого елементу “Z”

Z(max)=	
Xi(n)=	

X	Z
1	0,631578947
-1,4	3,031578947
3	1,368421053
1,5	0,131578947
1,1	0,531578947
1,5	0,131578947
4,9	3,268421053

Таблиця 3.7 – Завдання сьомого кроку.

Вірна відповідь буде такою: (таб. 3.8)

z(max)=	3,268421053
xi(n)=	4,9
X	Z
1	0,631578947
-1,4	3,031578947
3	1,368421053
1,5	0,131578947
1,1	0,531578947
1,5	0,131578947
4,9	3,268421053

Таблиця 3.8 – Відповідь на завдання сьомого кроку.

Крок 8.

Далі перед студентом відкривається наступний крок завдання, де необхідно ввести вірну відповідь з клавіатури.

При формулі знаходження $T(n) = (x_i(n) - x(n)) / S(n)$, $x_i(n)$ - елемент вибірки x , $x(n)$ - середнє вибіркове, $S(n)$ - Вибіркове стандартне відхилення. Якою буде відповідь якщо $S(n) = 1.262$, $x(n) = 1.631$, $x_i(n) = 4.9$

Вірна відповідь – 2,590

Крок 9.

На цьому кроці студенту потрібно вписати числа у формулу так, щоб формула стала тотожно вірною. Якщо кількість елементів - $n = 19$, а $T(n) = 2,588$.

$$t_{(n-2)} = \frac{\sqrt{n-2}T(n)}{\sqrt{n-1-T^2(n)}} \quad (3.3)$$

Вірна відповідь буде ототожнювати рівність.

Крок 10.

Далі питання: «В табличних редакторах типу MS Excel Значення t_{kr} рахується за формулою: --/-- коли p це коефіцієнт значимості, а n - кількість елементів у вибірці»

- a) - $TINV(p;n-2) / СТЬЮДРАСПОБР(p;n-2)$;
- b) - $TINV(n;p-2) / СТЬЮДРАСПОБР(n;p-2)$;
- c) - $ABS(p(n-2)) / ABS(p;n-2)$.

Вірна відповідь – a)

Крок 11.

Питання: «Якщо $t(n-2) > t_{kr}$ то...»

- a) - завершуємо пошук аномального елемента, так як його знайдено;
- b) - додаємо знайдений аномальний елемент у вибірку і повторюємо дії знаходження наступного аномального елемента;

- с) - виключаємо знайдений аномальний елемент з вибірки і повторюємо дії знаходження наступного аномального елемента.

Вірна відповідь – с)

Крок 12.

Питання: «Якщо твердження $t(n-2) = -3,737704918 < t_{kr} = 1,74588$ є вірним то...»

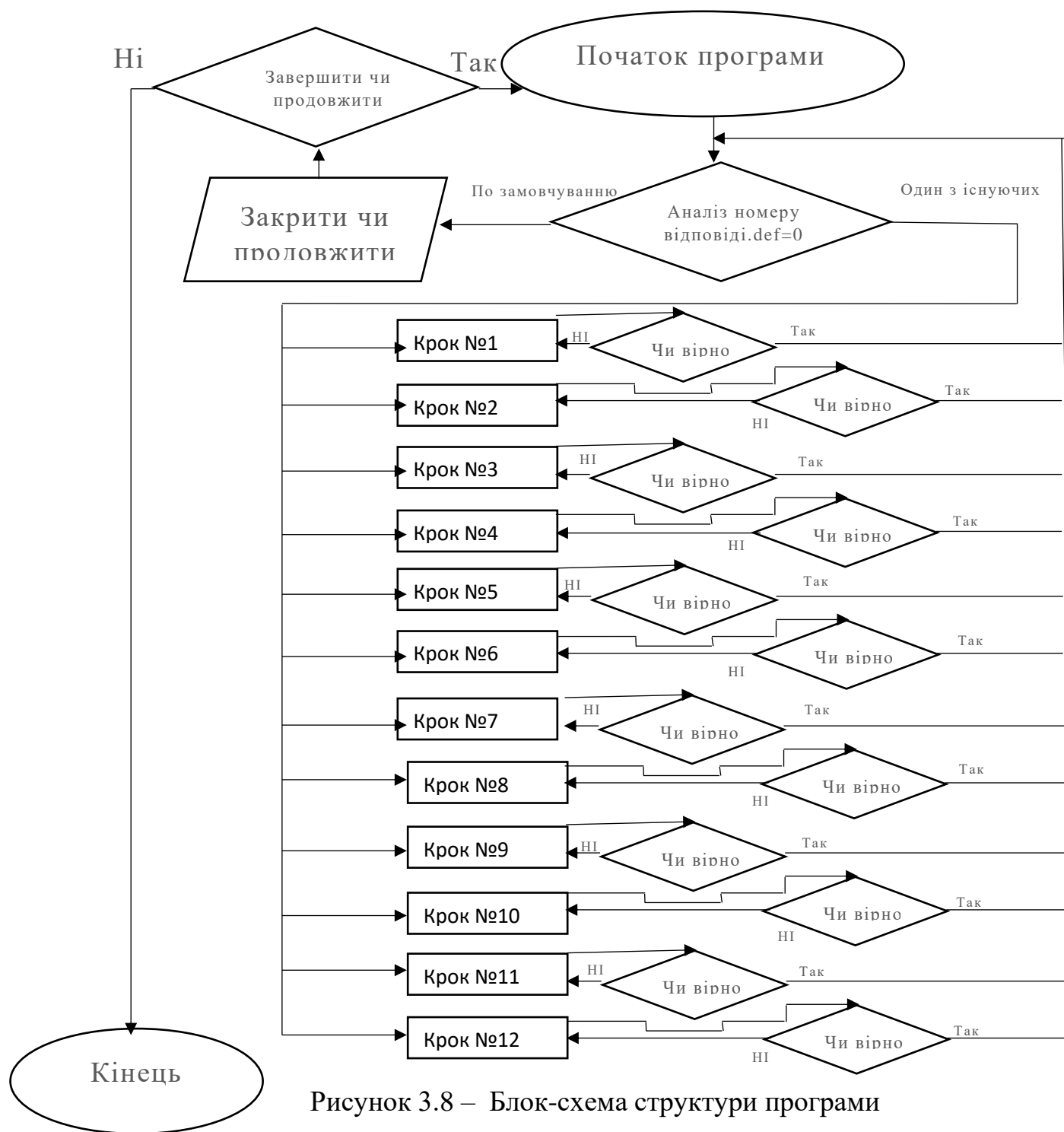
- а) робота методу завершується, так як підозрілий на аномальність елемент не є аномальним;
- б) робота методу продовжується, так як підозрілий на аномальність елемент не є аномальним;
- с) робота методу продовжується, так як підозрілий на аномальність елемент є аномальним.

Вірна відповідь – с)

В кінці роботи програми виводиться інформація про закінчення роботи Методу.

3.3 Розробка блок-схеми, яка підлягає програмуванню

Наведемо блок-схему, що дає уявлення про структуру програми, яка відповідає за вибір номеру питання.



Також наведемо блок-схему роботи програми для пунктів з вибором даних (рис 3.9) :

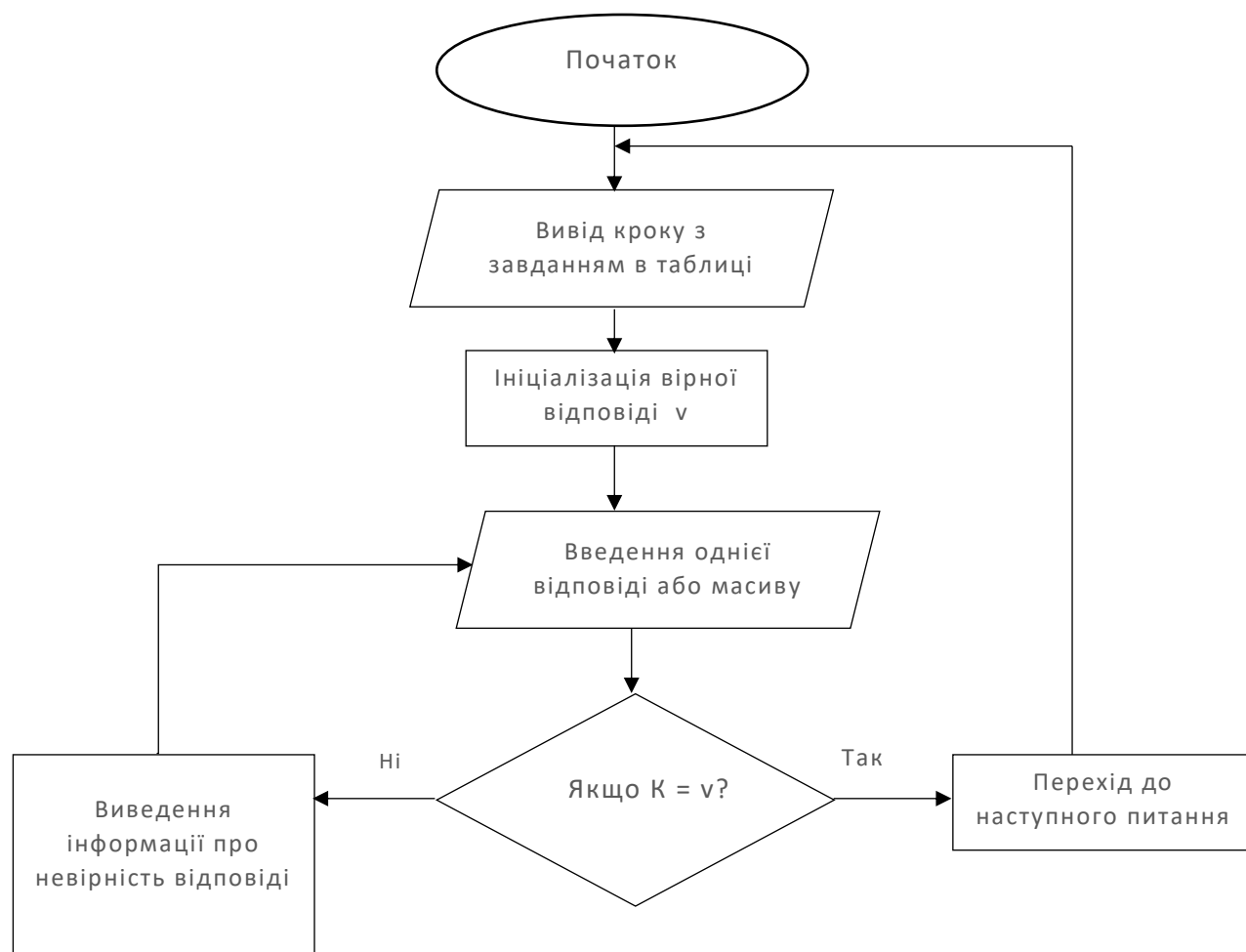


Рисунок 3.9 – Блок-схема роботи програми для кроків з введенням даних в таблицю

Наведемо блок-схему роботи підпрограми для пунктів з введенням даних з списку, що випадає (рис 3.10).

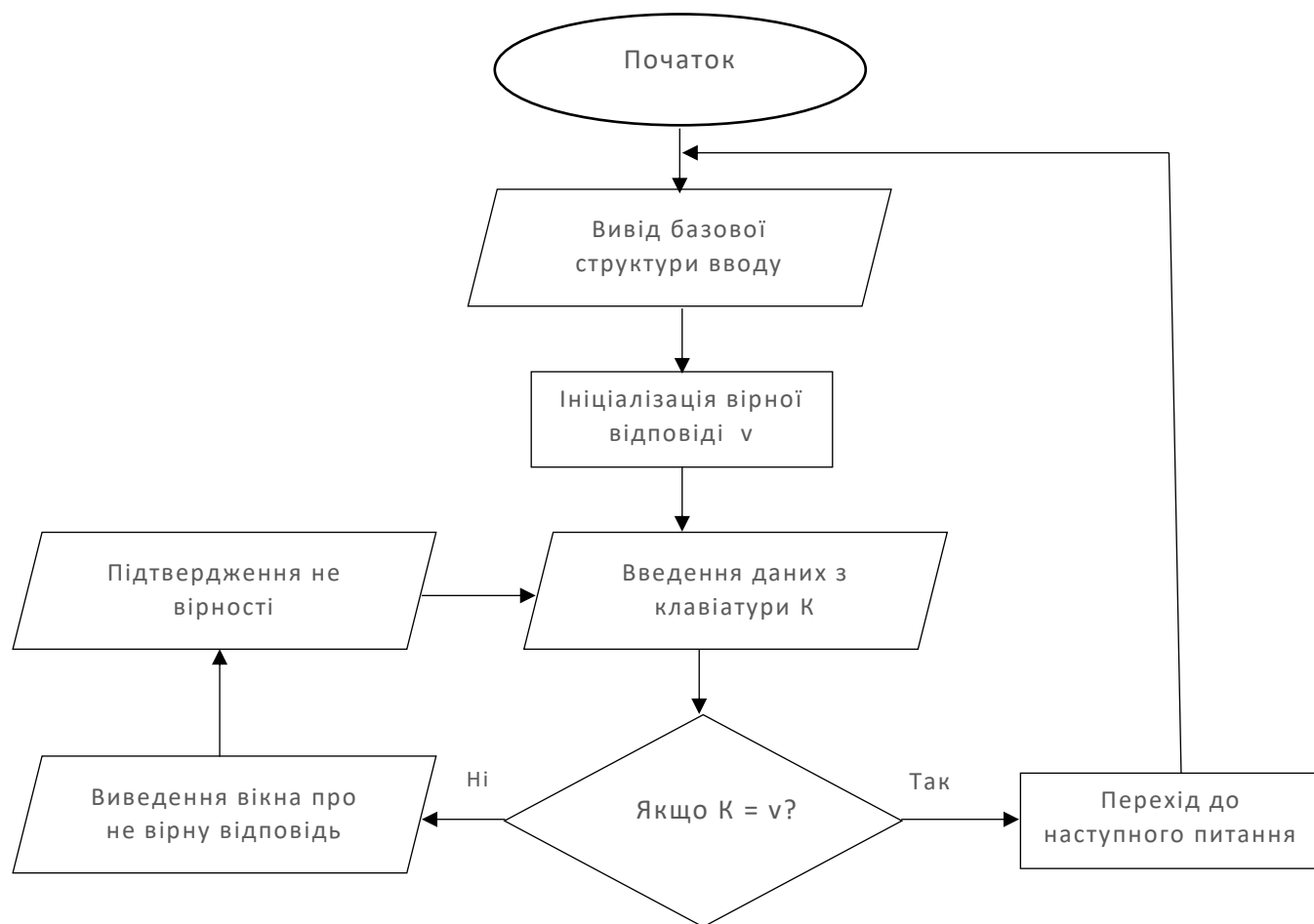


Рисунок 3.10 – Блок-схема роботи підпрограми для пунктів з вибором даних з таблиці

Наведемо блок-схему роботи підпрограми для перевірки введених даних (рис 3.11).

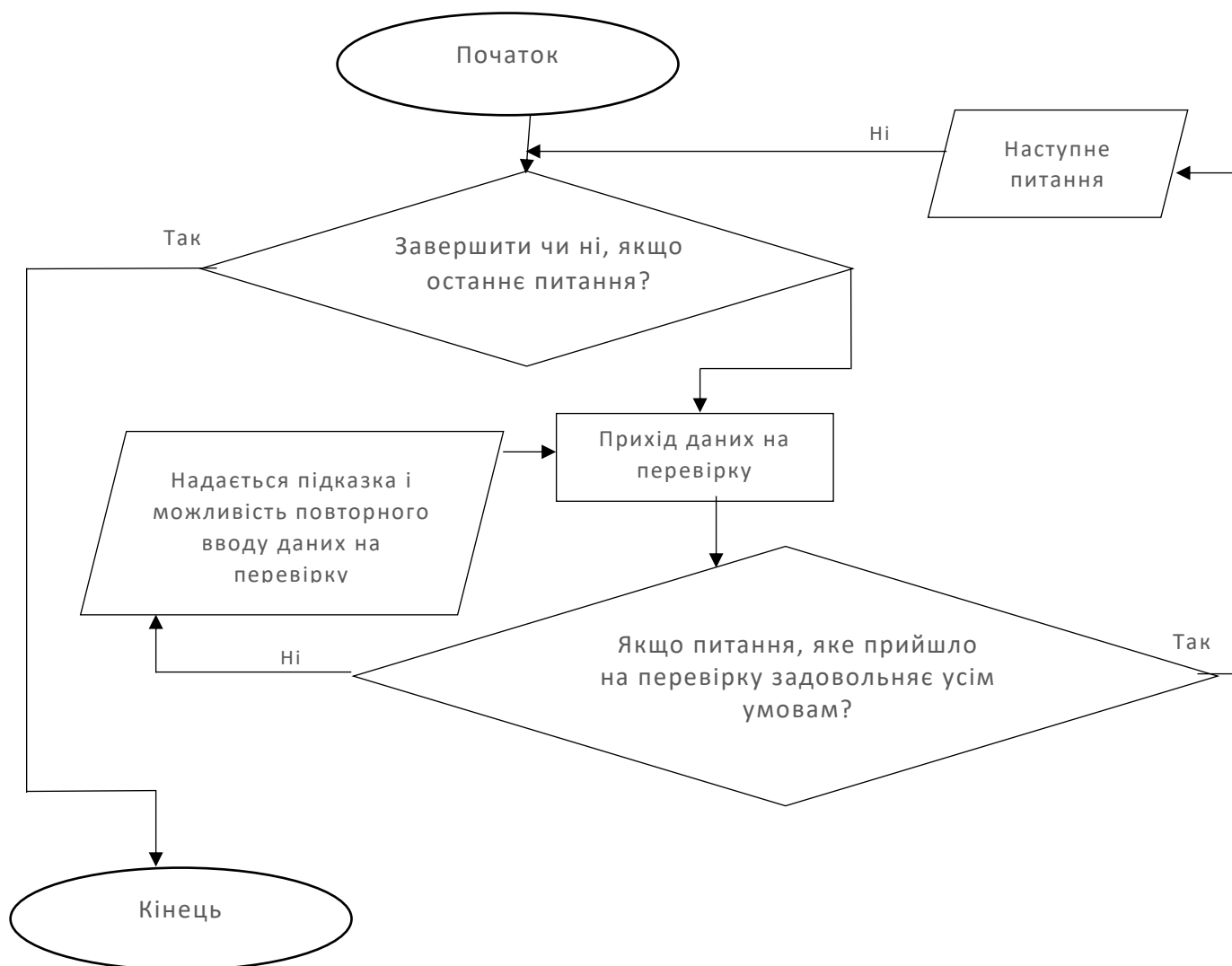


Рисунок 3.11 – Блок-схема роботи підпрограми для перевірки введених даних

3.4 Обґрунтування вибору програмних засобів для реалізації завдання роботи

Для реалізації тренажера було обрано мову програмування Javascript. Це рішення було прийняте тому, що в даний час більшість людства використовують свій комп'ютер в якості інструменту для доступу до мережі Інтернет, а тому «WEB-інтерфейс» є більш зручним та зрозумілим для користувачів. І тому для написання коду для «веб-застосунку» було обрано найпопулярнішу мову програмування в «WEB» - Javascript. Також однією з переваг цієї мови є її крос-платформеність, що дозволяє тренажеру працювати на будь-якій операційній системі не залежно від її архітектури та типу. Ще одним позитивним аспектом цієї мови є велика кількість бібліотек, що в свою чергу дозволило створити та скомпілювати тренажер не тільки, як «веб-застосунок», а й як повноцінний файл запуску з розширенням «.exe»

В процесі розробки тренажера було використано найбільш розповсюджене середовище розробки Visual Studio Code.

Основою вибору Visual Studio Code, як головного середовища розробки, було те, що він має можливість крос-платформенного запуску (робота під Linux), а також його зручність. Visual Studio Code є наслідником вільного проекту Atom (середовище розробки), що розвивається компанією GitHub. Зокрема, Visual Studio Code є надбудовою над Atom, який використовує браузерний рушій Chromium та Node.js. Редактор містить вбудований завантажувач, інструменти для роботи з Git, а також засоби рефакторингу, навігації по коду, автодоповнення типових конструкцій. Дане середовище розробки підтримує платформу Node.js, і позиціонує себе, як легковагове рішення, що дозволяє обійтися без повного інтегрованого середовища розробки.

Для надійності розробки було обрано систему контролю версій Git. Git є однією з найефективніших та найнадійніших і високопродуктивних систем керування версіями, що надає зручні засоби нелінійної розробки, які базуються на відгалуженні і злитті гілок різних версій проекту. Для забезпечення цілісності історії та стійкості до змін заднім числом використовуються криптографічні методи, також можлива

прив'язка цифрових підписів розробників до тегів і комітів. Ця система спроектована як набір програм, спеціально розроблених з врахуванням використання у скриптах, що дає можливість зручно створювати спеціалізовані системи керування версіями або користувацькі інтерфейси.

Створення та компілювання «веб- застосуноку» в «.exe» файл відбулося завдяки програмній платформі Node.js, а саме фреймворку Electron. Electron — фреймворк, розроблений GitHub, який дозволяє розробляти графічні застосунки для настільних операційних систем за допомогою веб-технологій.

4. ПРАКТИЧНА ЧАСТИНА

4.1 Опис процесу програмної реалізації

Роботу розпочато з створення нового проекту в VS-Code (рис 4.1).

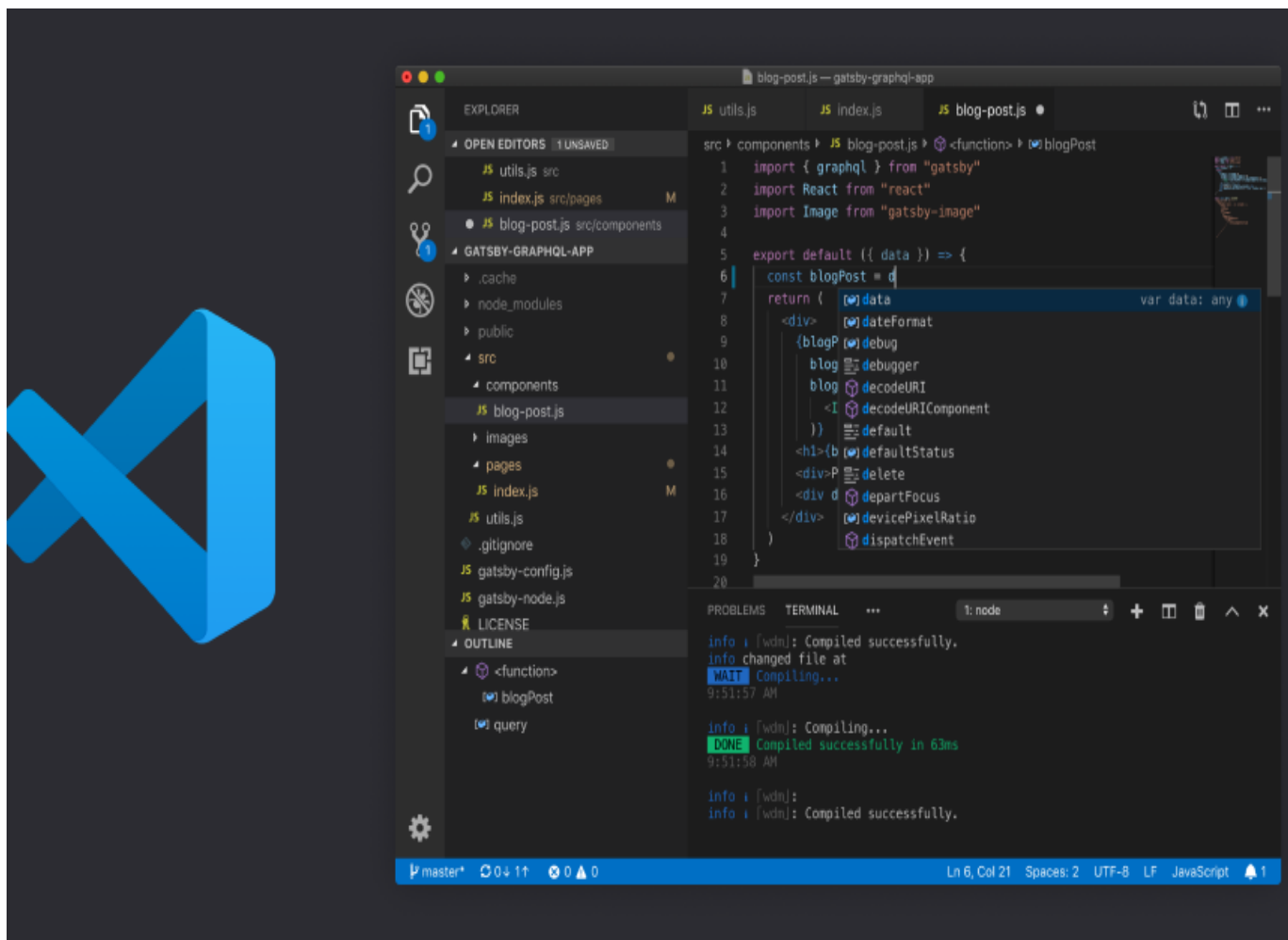


Рисунок 4.1 – Блок-схема роботи підпрограми для перевірки введених даних

Так, як в тренажері проводиться перевірка завдань на вірність, перевірку було вирішено зробити через умовний оператор Switch...case. Виконання реалізоване за допомогою функціонального типу програмування, а також мови розмітки HTML5 та каскадної таблиці стилів CSS.


```

1 function getAnswer(quest, tru_false) {
2
3     switch (quest) {
4         case 1:
5             if (tru_false == 1) {
6                 alert2(true, 'Все вірно!!!');
7                 startNextQuest(quest + 1);
8             } else {
9                 alert2(false, 'Не вірно, спробуйте ще раз.');
```

Перехід між питаннями також використовує комбінацію Switch...case.

```

1 function startNextQuest(questCount) {
2     getMainElement("quest_js").innerHTML = '';
3     switch (questCount) {
4         case 1:
5             uploadMainHtml(questCount);
6             uploadQuestTable("<h4>Виберіть вірну відповідь</h4>", "<h3> Аномальним
7 спостереженням називається? </h3>", " <div class='form-check'> <label class='form-
8 check-label'> <input type='radio' onclick='getAnswer(" + questCount +
9             ",1)' class='form-check-input' name='optradio'>-називається знаходження
10 та ідентифікація елементів, подій або спостережень, що не відповідають очікуваній
11 поведінці або іншим елементам набору даних;</label></div><div class='form-check'>
12 <label class='form-check-label'> <input type='radio' onclick='getAnswer(" +
```

```

13         questCount +
14         ",0)" class='form-check-input' name='optradio'>-називається знаходження
15 та ідентифікація елементів, подій або спостережень, що відповідають очікуваній
16 поведінці (патернам) або іншим елементам набору даних;</label></div><div
17 class='form-check disabled'> <label class='form-check-label'> <input
18 type='radio' onclick='getAnswer(" +
19         questCount + ",0)" class='form-check-input' name='optradio'>-називається
20 знаходження відповідей між очікуваними подіями та набором даних суми всіх
21 елементів.</label></div> ");
22     break;

```

Інформаційний банер виводиться за допомогою бібліотеки «SweatAlert2», а в частоті за допомогою функції `getAnswer`.

```

1 function getAnswer(quest, tru_false) {
2
3     switch (quest) {
4         case 1:
5             if (tru_false == 1) {
6                 alert2(true, 'Все вірно!!!');
7                 startNextQuest(quest + 1);
8             } else {
9                 alert2(false, 'Не вірно, спробуйте ще раз.');
```

Завантаження завдання для наступного кроку тренажера відбувається за допомогою функції `uploadQuestTable`.

```

1 function uploadQuestTable(htmlWithText1, htmlWithText2, htmlWithText3) {
2     getMainElement("main-card-quest").appendChild(addElements("<div class='table-
3 active' ><table class='table' style='table-layout: fixed; color: whitesmoke
4 !important; width: 100%;'><tbody><tr><td>" + htmlWithText1 +

```

```

    "</br></br></br></br></br></td><td style='border-left:1px solid white'>" +
    htmlWithText2 + "</td><td style='border-left:1px solid white'>" + htmlWithText3));
}

```

Підключення калькулятора до тренажеру виконано за допомогою функції `getCalc`.

```

function getCalc(val) {
    Swal.fire({
        title: '<strong>Калькулятор</strong>',
        html: '<iframe src="./jscalc/index.html" width="300" height="410"
1 style="border:none;overflow:hidden" scrolling="no" frameborder="0"
2 allowTransparency="true" allow="encrypted-media"></iframe>',
3
4 showCloseButton: true,
5 focusConfirm: false,
6 confirmButtonText:
7     '<i class="fa fa-thumbs-up"></i> ok!',
8 confirmButtonAriaLabel: 'Закрити',
9 position: 'top-start'
10
11 })
12
13 $(".swal2-modal").css('background-color', 'rgb(155, 136, 226)'); //Optional
14 changes the color of the sweetalert
15 $(".swal2-container").css('background-color', 'rgba(232, 232, 232, 0.8)');
16 //changes the color of the overlay
17 $(".swal2-container").css('background', 'solid black'); //changes the color of
18 the overlay
    $(".swal2-content").css('color', 'white'); //changes the color of the overlay
}

```

Дизайн тренажеру CSS, використовує прив'язку до так званого `id` елементу.

```

#body {
    display: -ms-flexbox;
    display: -webkit-box;
    display: flex;
    -ms-flex-align: center;
    -ms-flex-pack: center;
    -webkit-box-align: center;
    align-items: center;
    -webkit-box-pack: center;
    justify-content: center;
    padding-top: 40px;
    padding-bottom: 40px;
    background-color: #f5f5f5;
}

```

4.2 Опис програми

На (рис4.2) та (рис4.3) вказано вигляд головного вікна тренажера. На якому розміщені головні дані про тренажер, а також дані про метод, якому присвячений тренажер.

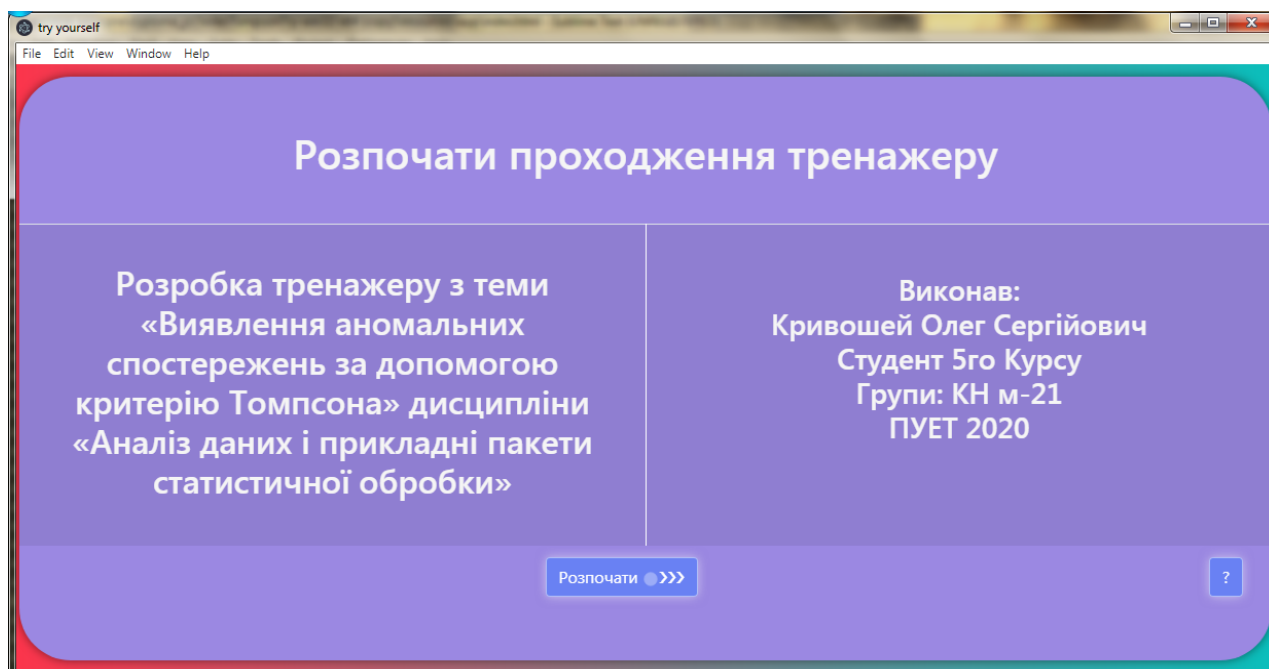


Рисунок 4.2 – Головне вікно тренажера



На Рисунок 4.3 – Дані про метод (рис4.4)

знаходиться вигляд вікна тренажера з кроком №1. Також на (рис4.5), (рис4.6) розміщено вигляд вікна тренажера в момент вірної або не вірної відповіді.

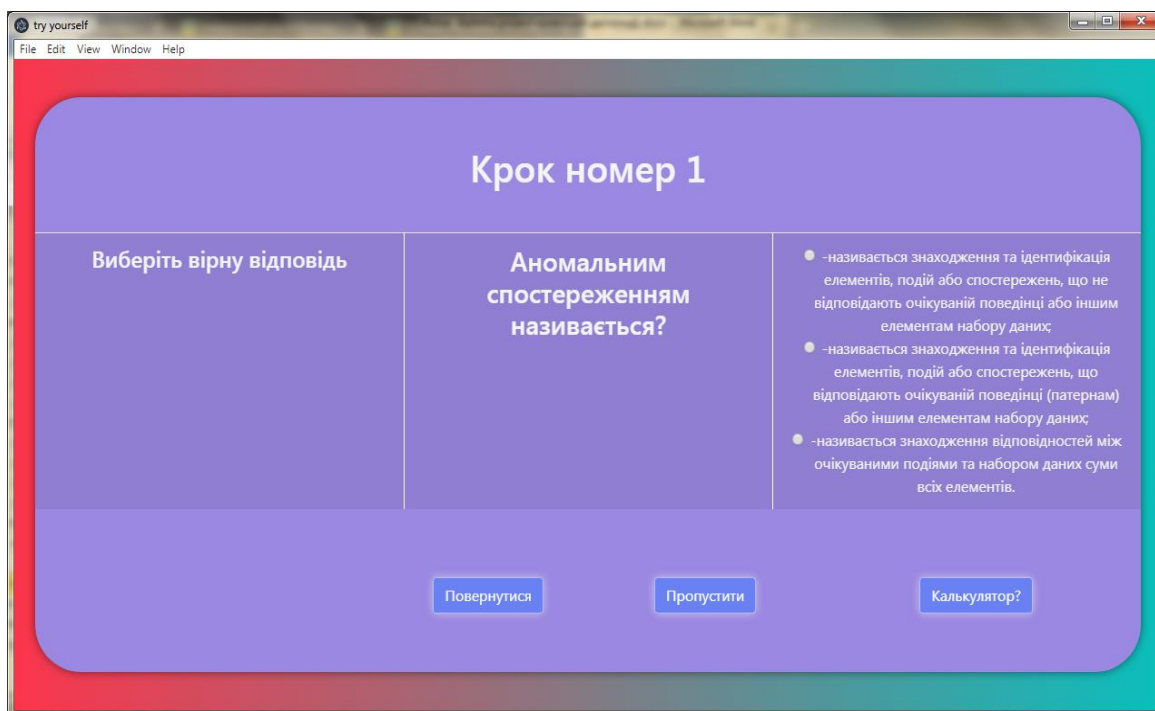


Рисунок 4.4 – Крок 1

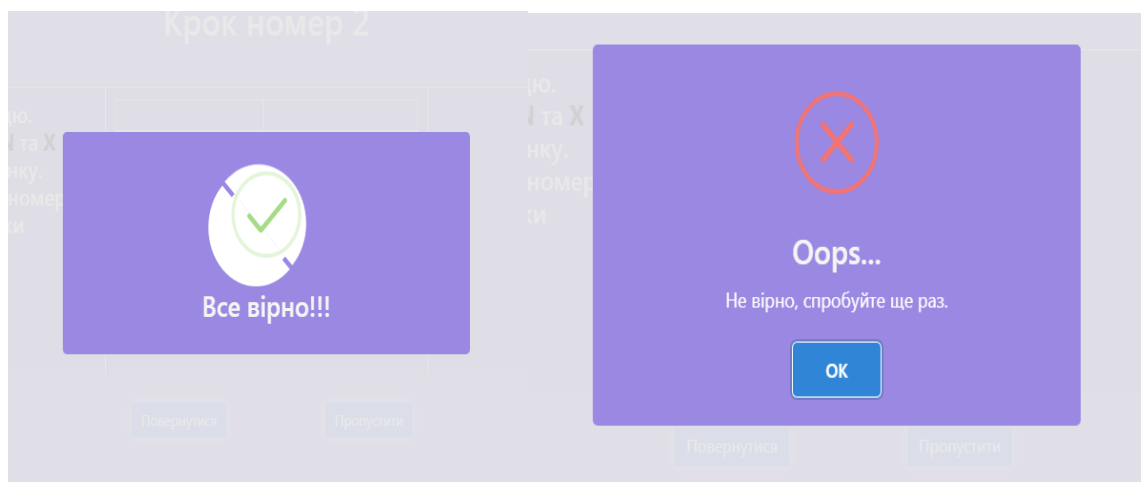


Рисунок 4.5 – Момент
вірної відповіді

Рисунок 4.6 – Момент не
вірної відповіді

На (рис4.7) розміщено вигляд вікна тренажеру з калькулятором. Калькулятор додано в тренажер для зручності підрахунку під час роботи програми на тих кроках, де потрібно вводити розраховані данні в таблицю. Також, на (рис4.8) вказано крок №2, в якому необхідно, щоб студент ввів дві вірні відповідні в пропущені комірки таблиці.

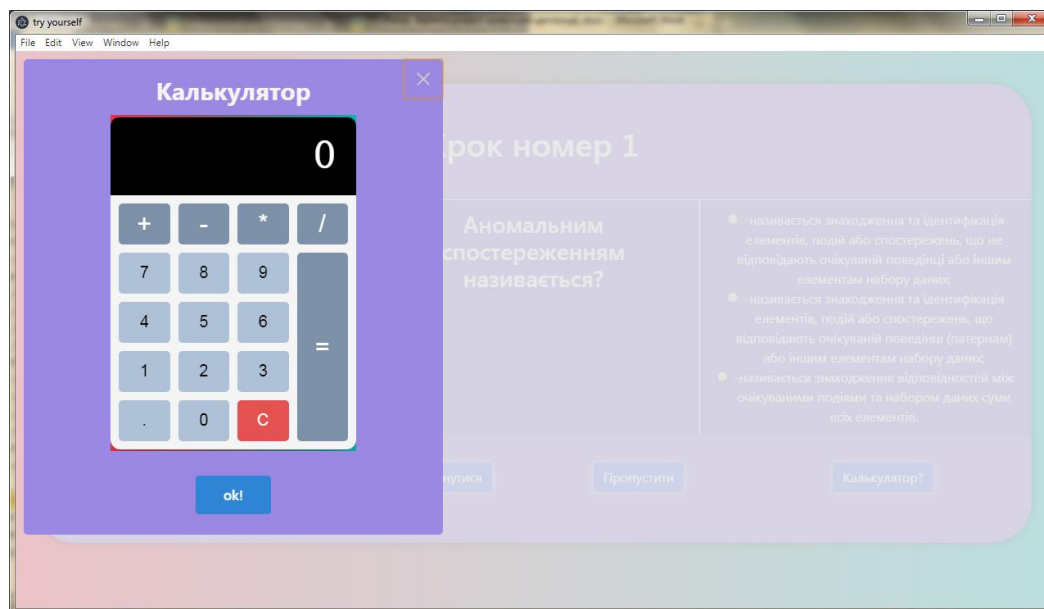


Рисунок 4.7 – Вбудований калькулятор

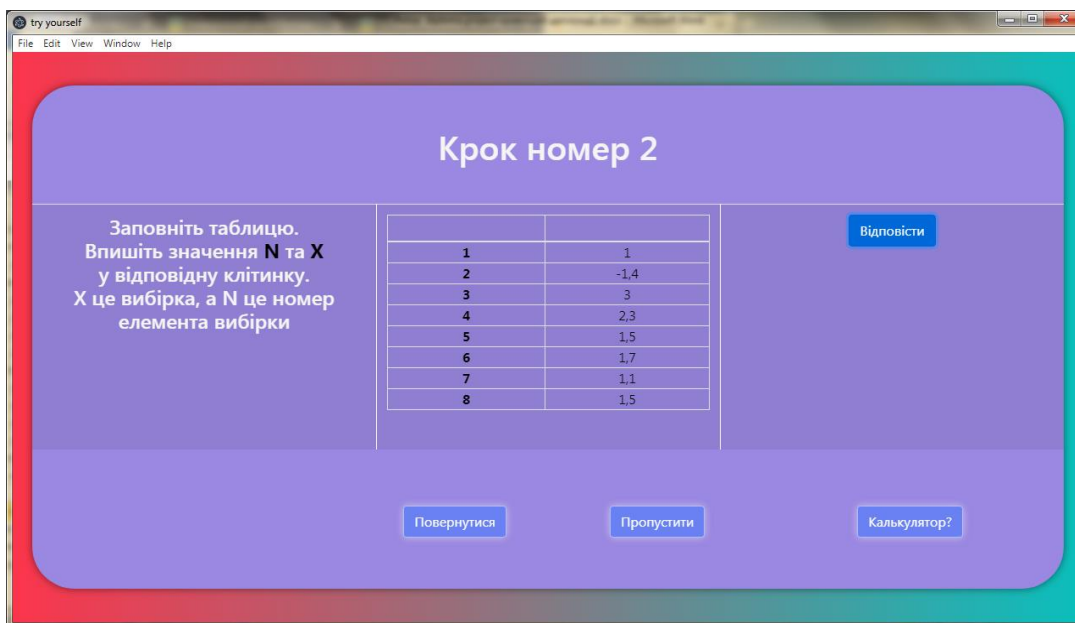


Рисунок 4.8 – Крок 2

На (рис4.9) розміщено вигляд вікна тренажеру на кроці №3. На (рис4.10) розміщено підказку, яка використовується на кроці №3.

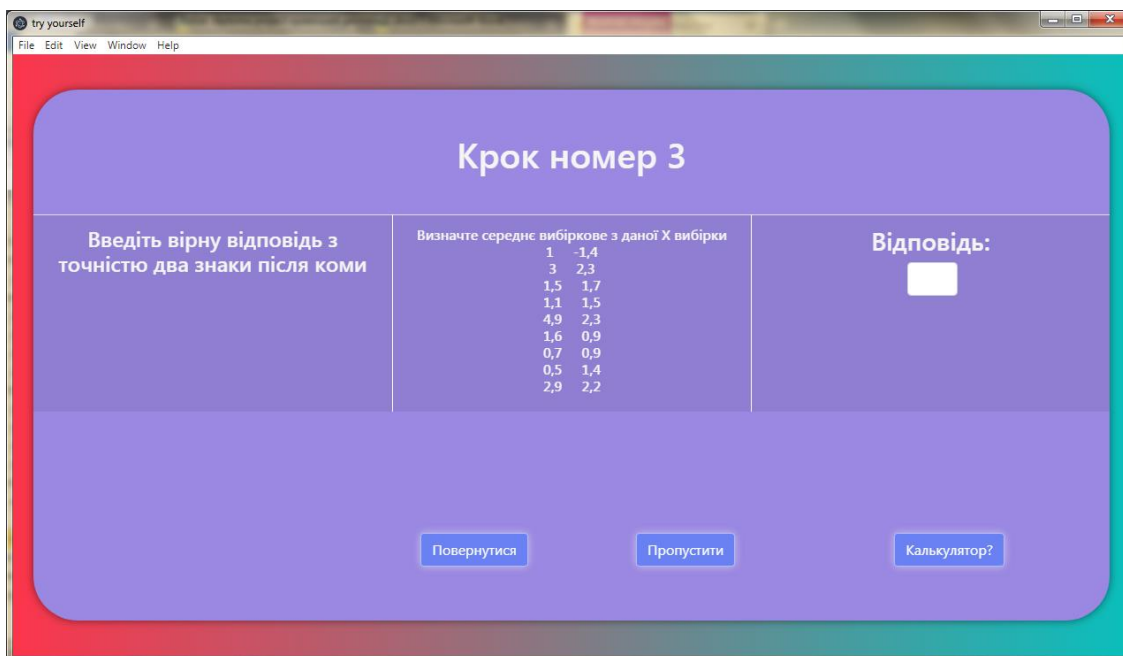


Рисунок 4.9 – Крок 3

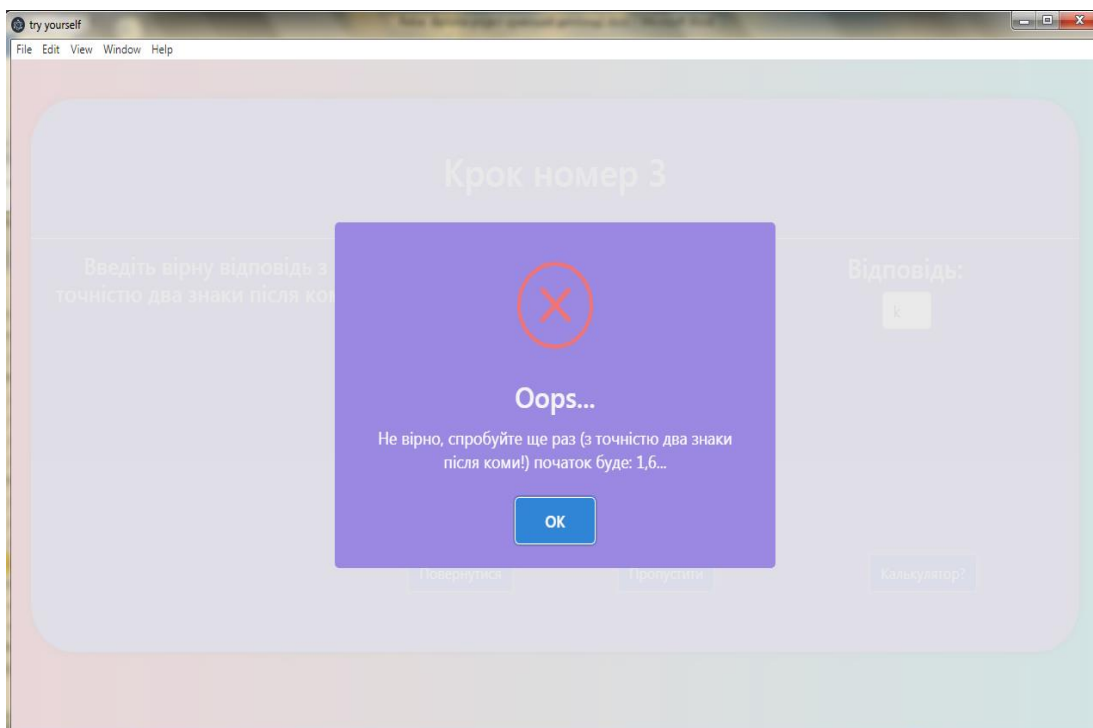


Рисунок 4.10 – Підказка на кроці №3

На (рис4.11) та (рис4.12) розміщено вигляд вікна тренажеру на кроці №4 та кроці №5.

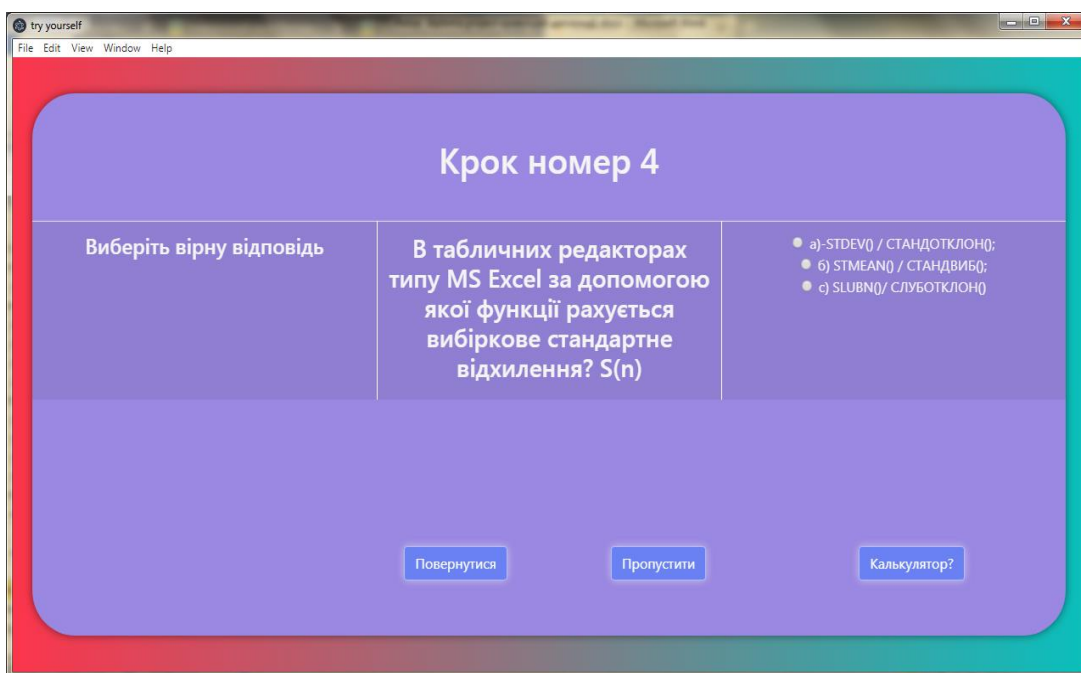


Рисунок 4.11 – Крок 4

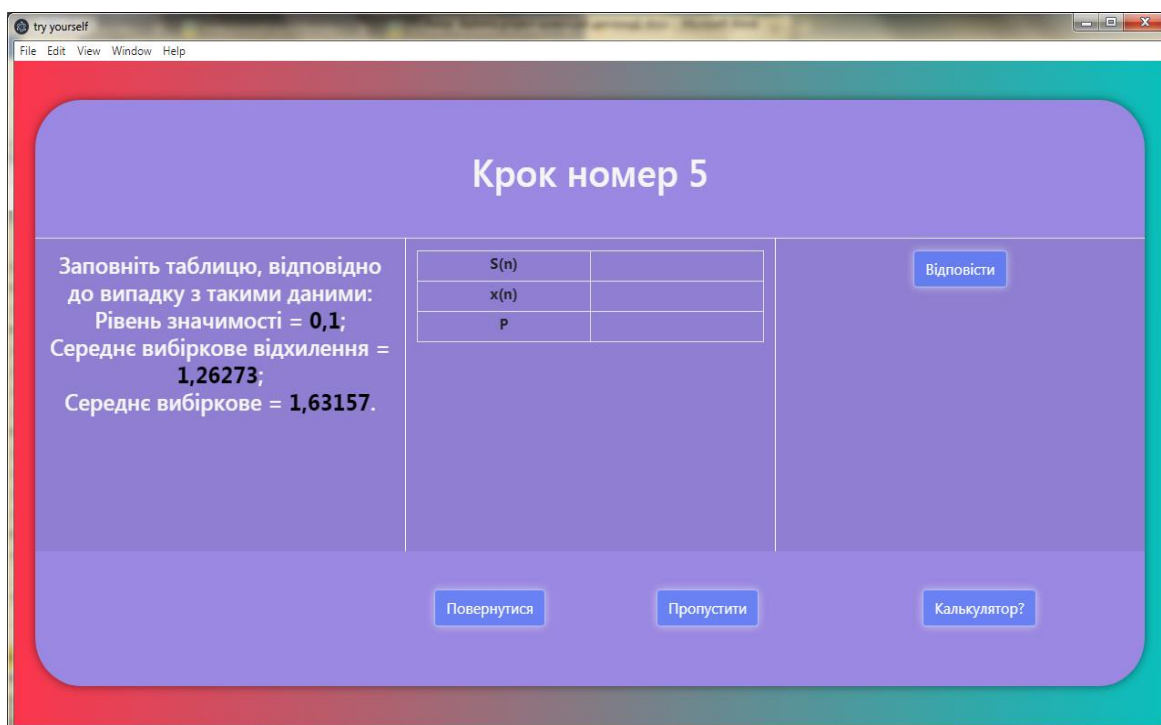


Рисунок 4.12 – Крок 5

На (рис4.13) та (рис4.14) розміщено вигляд вікна тренажеру на кроці №6 та №7.

try yourself

File Edit View Window Help

Крок номер 6

Вкажіть пропущені число в стовпчик "Z", враховуючи що кожен елемент "Zi" знаходиться за формулою $Z_i = ABC(x_i - \bar{x} (n))$

Середнє вибіркове		1,631
n	x	z
1	1	0,631
2	-1,4	3,031
3	3	
4	2,3	0,668
5	1,5	0,131

Відповісти

Повернутися Пропустити Калькулятор?

Рисунок 4.13 – Крок 6

try yourself

File Edit View Window Help

Крок номер 7

Виберіть найбільше число в стовбчику "Z" та впишіть його в пусту клітинку "Z(max)", в пусту клітинку "Xi(n)" впишіть елемент "Xi" який знаходиться напроти найбільшого елемента "Z"

Z(max)=	
Xi(n)=	
X	Z
1	0,631578947
-1,4	3,031578947
3	1,368421053
1,5	0,131578947
1,1	0,531578947
1,5	0,131578947
4,9	3,268421053

Відповісти

Повернутися Пропустити Калькулятор?

Рисунок 4.14 – Крок 7

На (рис4.15) та (рис4.16) розміщено вигляд вікна тренажеру на кроці №8 та №9 де необхідно вводити данні з клавіатури.

try yourself

File Edit View Window Help

Крок номер 8

Введіть вірну відповідь	<p>При формулі знаходження $T(n) = (x_i(n) - \bar{x}(n)) / S(n)$, $x_i(n)$ - елемент вибірки x, $\bar{x}(n)$ - середнє вибіркове, $S(n)$ - вибіркове стандартне відхилення</p> <p>якою буде відповідь якщо $S(n) = 1.262$, $\bar{x}(n) = 1.631$, $x_i(n) = 4.9$</p>	<p>Відповідь:</p> <input type="text"/>
-------------------------	---	--

Повернутися Пропустити Калькулятор?

Рисунок 4.15 – Крок 8

try yourself

File Edit View Window Help

Крок номер 9

Впишіть числа так, щоб формула стала тотожно вірною	<p>$t_{(n-2)} = \frac{\sqrt{n-2} T(n)}{\sqrt{n-1 - T^2(n)}}$</p> <p>Якщо кількість елементів - $n = 19$, а $T(n) = 2,588$</p>	<p>Відповідь:</p> <p>$t(n-2) = (\text{ } - 2)^{0,5} * \text{ } / (\text{ } - 1 - \text{ } ^{0,5} = 3,1747$</p> <p>Відповісти</p>
---	--	---

Повернутися Пропустити Калькулятор?

Рисунок 4.16 – Крок 9

На (рис4.17) та (рис4.18) розміщено вигляд вікна тренажеру на кроці №10 та №11, які виконані у формі вибору вірної відповіді зі списку.

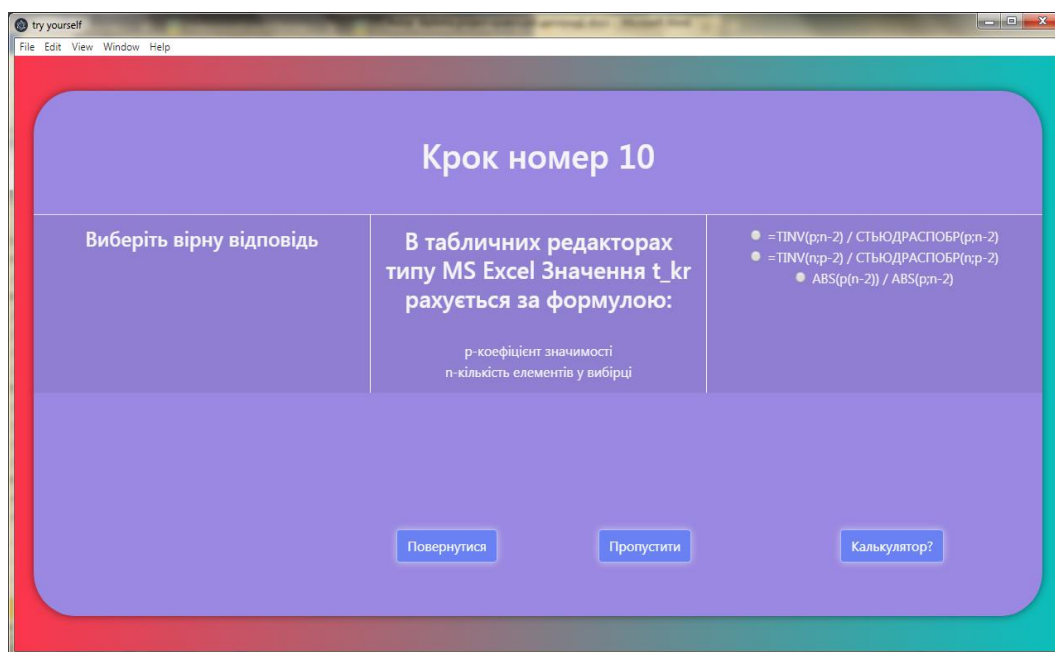


Рисунок 4.17 – Крок 10

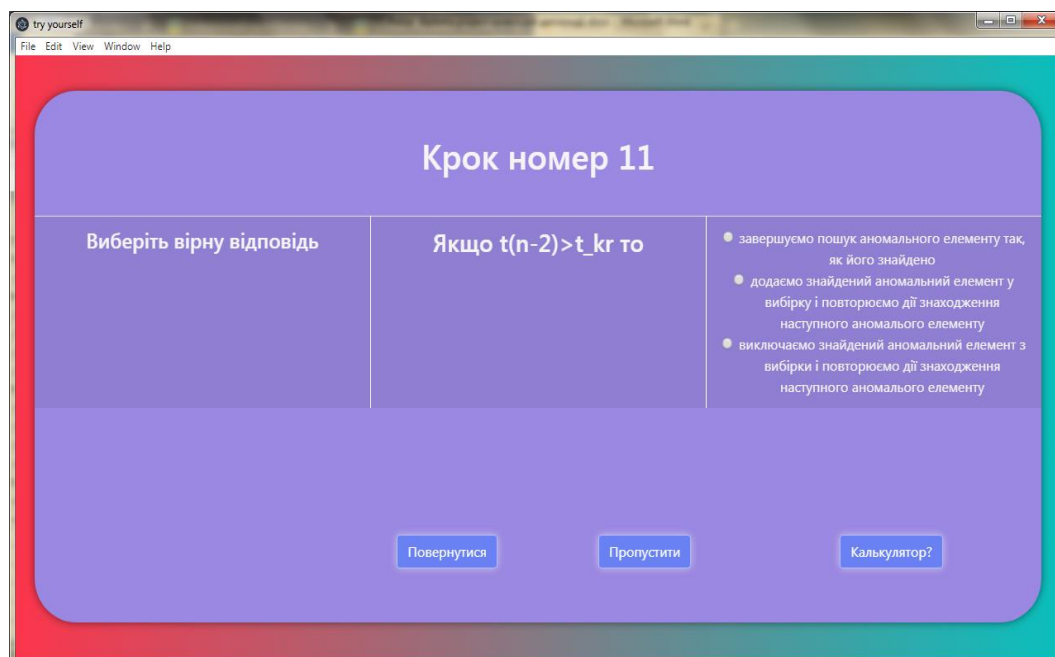


Рисунок 4.18 – Крок 11

На (рис4.19) та (рис4.20) розміщено вигляд вікна тренажеру на кроці №12 та інформаційний банер про завершення методу.

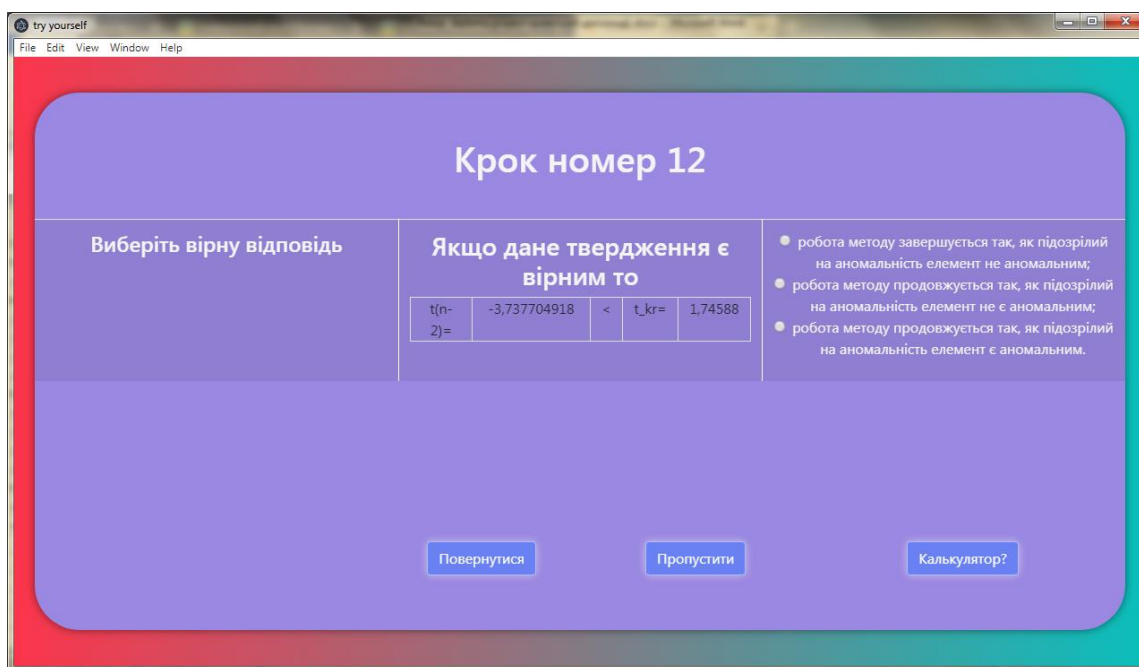


Рисунок 4.19 – Крок 12

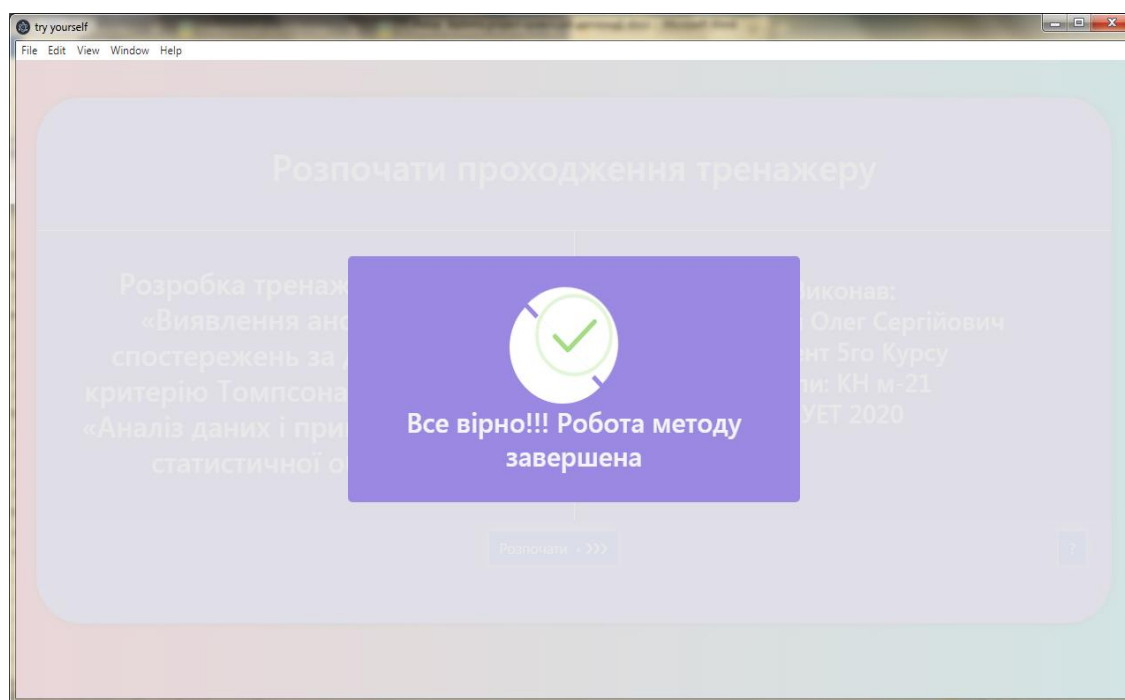


Рисунок 4.20 – Банер про завершення методу

ВИСНОВКИ

Впровадження інформаційних технологій в процес навчання дозволяє ефективно підвищити якість освіти та якість отримання вмінь та навичок студентів за допомогою віртуальних програм-тренажерів. Тренажер створений на основі інноваційних рішень в комп'ютерних технологіях. Він забезпечує підготовку студентів до подальшого вирішення проблем при розв'язуванні задач з виявлення аномальних спостережень за допомогою критерію Томпсона. Також під час виконання магістерської роботи було проаналізовано літературні джерела та програми-тренажери студентів минулих років.

Поставлена мета була досягнута: було розроблено тренажер з теми «Виявлення аномальних спостережень за допомогою критерію Томпсона» дисципліни «Аналіз даних і прикладні пакети статистичної обробки».

Результатами дипломної роботи є:

1. Обрано середовище та мова програмування.
2. Розроблено алгоритм тренажера з теми «Виявлення аномальних спостережень за допомогою критерію Томпсона» дисципліни «Аналіз даних і прикладні пакети статистичної обробки».
3. Програмно реалізовано тренажер та відлагоджено програму на відсутність помилок та багів.

Позитивними моментами розробленого програмного продукту є:

1. Інтуїтивно зрозумілий інтерфейс тренажера та його сучасний дизайн.
2. Під час кожного моменту введення даних передбачена перевірка даних та, у випадку невірності відповіді, відбудеться виведення банера про невірність введених даних, а також надається підказка.
3. Програма розроблена під платформу Chromium, тож вона може бути запущена, як на будь-якій операційній системі, так і влаштована на сайт, як одна із його компонентів.
4. Програма розроблена на інтерпретованій мові програмування.

Всі стилі програми було прописано у вигляді окремого елементу програми «Style». Також було створено відповідні функції для опрацювання та виведення вмісту кожного з кроків алгоритму тренажеру. Щоб виводити вміст, в залежності від номеру теми, реалізовано функцію `startNextQuest`. Для роботи всіх кнопок було розроблено події. Всі вимоги, описані в постановці задачі, були виконані.

Результати роботи можуть бути використані при вивченні студентами відповідної теми з дисципліни «Аналіз даних і прикладні пакети статистичної обробки».

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Ємець О. О. Про розробку тренажерів для дистанційних курсів кафедрою ММСІ ПУЕТ // Інформатика та системні науки (ІСН-2017) [Електронний ресурс]: матеріали VIII Всеукраїнської науково-практичної конференції за міжнародною участю, (м. Полтава, 16-18 березня 2017 р.) / за ред. О. О. Ємець. – Полтава: ПУЕТ, 2017 –С. 152-161. Режим доступу: <http://dspace.puet.edu.ua/handle/123456789/2616>.
2. Мазур М.П. Розвиток дистанційного навчання в Україні як складової інформатизації сучасного суспільства / М.П.Мазур // Інформатика та інформаційні технології в навчальних закладах. –2007, No1 . –с. 71-75
3. Безмєнов Е. Г. Розробка програмного забезпечення для тренажера з теми «Неоднорідні диференціальні рівняння другого порядку із сталими коефіцієнтами» з дисципліни «Математичний аналіз»/ Е. Г. Безмєнов // КОМП'ЮТЕРНІ НАУКИ І ПРИКЛАДНА МАТЕМАТИКА (КНіПМ-2018): матеріали науково-практичного семінару. Випуск 1 /за ред. Ємця О.О. –Полтава: Кафедра ММСІ ПУЕТ, 2018. –45 - 54 с.Режим доступу -<http://dspace.puet.edu.ua/handle/123456789/6979>.
4. КОМП'ЮТЕРНІ НАУКИ І ПРИКЛАДНА МАТЕМАТИКА (КНіПМ-2018): матеріали науково-практичного семінару. Випуск 2 / за ред. Ємця О.О. –Полтава: Кафедра ММСІ ПУЕТ, 2018. –27 с. Режим доступу - <http://dspace.puet.edu.ua/handle/123456789/6987>.
5. Кильник В. В. Програмна реалізація елементів тренажеру з теми «Навчання елементарного персептрону» дисципліни «Нейронно-мережеві технології в інформатиці» / В. В. Кильник, Ю. Ф. Олексійчук // Комп'ютерні науки і прикладна математика (КНіПМ-2018): матеріали науково-практичного семінару. Випуск 1 – Полтава: Кафедра ММСІ ПУЕТ, 2018. – С. 54-58.
6. Олексійчук Ю. Ф. Розробка та впровадження дистанційного курсу з дисципліни «Програмування» / Ю. Ф. Олексійчук // Дистанційна освіта: забезпечення доступності та неперервної освіти впродовж життя (e-learning and university

- education-2017): матеріали XLII Міжнародної науково-методичної конференції (м. Полтава, 9–10 лютого 2017 року) – Полтава: ПУЕТ, 2017. – С. 167-169.
7. Олексійчук Ю. Ф. Програмна реалізація елементів тренажеру з теми «Сортування бульбашками» дисципліни «Аналіз алгоритмів» / Ю. Ф. Олексійчук,
 8. В. О. Голубенко // Комп'ютерні науки і прикладна математика (КНіПМ-2018): матеріали науково-практичного семінару. Випуск 1 – Полтава: Кафедра ММСІ ПУЕТ, 2018. – С. 11-16.
 9. Мандрика В. М. Тренажер з теми «1-R алгоритм» дисципліни «Комп'ютерний аналіз статистичних даних» / В. М. Мандрика, Ю. Ф. Олексійчук // Комп'ютерні науки і прикладна математика (КНіПМ-2018): матеріали науково-практичного семінару. Випуск 1 – Полтава: Кафедра ММСІ ПУЕТ, 2018. – С. 27-30.
 10. Русін В. С. Програмна реалізація елементів тренажеру з теми "Аналіз алгоритму сортування вставками" дисципліни "Аналіз алгоритмів" / В. С. Русін, Ю. Ф. Олексійчук // Інформатика та системні науки (ІСН-2017): матеріали VIII Всеукраїнської науково-практичної конференції за міжнародною участю (м. Полтава, 16–18 березня 2017 р.) – Полтава: ПУЕТ, 2017. – С. 236-237.
 11. Ярмоленко А. В. Алгоритм роботи тренажеру з теми «Асимптотичні оцінки функцій» дисципліни «Аналіз алгоритмів» / А. В. Ярмоленко, Ю. Ф. Олексійчук // Комп'ютерні науки і прикладна математика (КНіПМ-2018): матеріали науково-практичного семінару. Випуск 2 – Полтава: Кафедра ММСІ ПУЕТ, 2018. – С. 14-16.
 12. Голубенко В. О. Програмна реалізація тренажеру з теми «Сортування бульбашками» дисципліни «Аналіз алгоритмів» / В. О. Голубенко, Ю. Ф. Олексійчук // Комп'ютерні науки і прикладна математика (КНіПМ-2018): матеріали науково-практичного семінару. Випуск 2 – Полтава: Кафедра ММСІ ПУЕТ, 2018. – С. 6-9.

13. Спорнік С. С. Алгоритм тренажера з теми «Частинна кореляція» дисципліни «Аналіз даних і прикладні пакети статистичної обробки» / С. С. Спорнік, Ю. Ф. Олексійчук // Актуальні питання розвитку науки та забезпечення якості освіти у XXI столітті: тези доповідей XLIII Міжнародної наукової студентської конференції за підсумками науково-дослідних робіт студентів за 2019 рік (м. Полтава, 07–08 квітня 2020 року). Частина 2 – Полтава: ПУЕТ, 2020 – С.127-129.
14. Фесенко Д. І. Розробка тренажера з теми «Аналіз алгоритму сортування злиттям» дисципліни «Аналіз алгоритмів» / Д. І. Фесенко, Ю. Ф. Олексійчук // Актуальні питання розвитку науки та забезпечення якості освіти у XXI столітті: тези доповідей XLIII Міжнародної наукової студентської конференції за підсумками науково-дослідних робіт студентів за 2019 рік (м. Полтава, 07–08 квітня 2020 року). Частина 2 – Полтава: ПУЕТ, 2020 – С.129-131.
15. Кривошей О.С. Оптимізація перевезень сільгосппродукції: програмна реалізація тренажера (моделювання та розв’язування) дистанційного курсу «Проектне навчання з дисципліни «Методи оптимізації та дослідження операцій» / О. С. Кривошей, О. О. Ємець // Комп’ютерні науки і прикладна математика (КНіПМ-2019): матеріали науково-практичного семінару. Випуск 2 – Полтава: Кафедра ММСІ ПУЕТ, 2019. – С.40-60.
16. Сивокінь О.Ю. Тренажер з теми: «Логіка висловлювань» дистанційного навчального курсу «Математична логіка» та розробка його програмного забезпечення / О. Ю. Сивокінь, О. О. Черненко // Математична логіка: матеріали науково-практичного семінару. – Полтава: Кафедра ММСІ ПУЕТ, 2020. – С.42-60.
17. Мельницький Я.В. «Пояснювальна записка до бакалаврської роботи на тему: «Розробка тренажеру для вивчення основ мови програмування python» / Я.В. Мельницький // Полтава: Кафедра ММСІ ПУЕТ, 2020. – 58 с.

18. Теорія і практика використання системи Moodle для дистанційного навчання /Н.В. Самсоненко, О.Г. Сидорина, Л.В. Павлович.//Конференція ПНПУ – 2017.– ст.1
19. Флэнаган Д., JavaScript Подробное руководство: Шестое издание. Классика Computer Science. – СПб: Символ Плюс, 2012. – 1080 с., ил.
20. Чілікіна Т. В. Огляд тренажерів з дисципліни "Математичний аналіз" на прикладі розробок студентів напряму "Інформатика" / Т. В. Чілікіна // Інформатика та системні науки (ІСН-2016): матеріали VII Всеукраїнської науково-практичної конференції за міжнародною участю, (м. Полтава, 10–12 берез. 2016 р.). –Полтава: ПУЕТ, 2016. –С. 329-330.
21. Берк К., Кэйри П. Анализ данных с помощью Microsoft Excel.: Пер. с англ. –М.: Издательский дом «Вильямс», 2005. –560 с.
22. Минько А.А. Статистический анализ в MS Excel. –М.: Издательский дом «Вильямс», 2004. –448 с.
23. Тюрин Ю.Н., Макаров А.А. Анализ данных на компьютере / По.ред. В.Э. Фигурнова. –М.: ИНФРА-М, Финансы и статистика, 1995. –384 с.
24. Бібліографічний запис. Бібліографічний опис. Загальні вимоги та правила складання: ДСТУ 7.1-2006. –[Чинний від 2007-07-01]. –К. : Держспоживстандарт України, 2007. –47 с.
25. Кантелон Майк, Мек Брэдлі, Янг Алекс. Node.js в действии. –СПб.: Издательство «Питер», 2014. –548 с.

ДОДАТОКИ

Додаток А. Компакт-диск з тренажером

Додаток Б. Код програми